

~~specified by said instruction, reads an operand value from either said first register set or second register set as specified by said instruction, and writes a result value to said first register set or said second register set as specified by said instruction.~~

- 2³². The system of claim ¹31, wherein said data of a first type is 32 bits.
- 3³³. The system of claim ¹31, wherein said data of a second type is 64 bits.
- 4³⁴. The system of claim ¹31, wherein said first and second register sets each have two write ports and five read ports.
- 5³⁵. The system of claim ¹31, wherein said execution unit operates on integer data.
- 6³⁶. The system of claim ¹31, wherein said instruction performs operations upon operands to generate results, each instruction specifying a respective source address for each operand and a destination address for said result of said instruction, each address specifying a register set and an offset.
- 7³⁷. The system of claim ¹31, wherein said data stored in said second register set has a size that is different from the size of the register in which the data is contained.
- 8³⁸. The system of claim ⁷37, wherein said execution unit stores integer data within said second register set.
- 9³⁹. The system of claim ⁷37, wherein said execution unit selects the size of operand to read from, or write to, a floating-point register based upon the type of instruction being executed rather than upon the width of the register from which the integer operand is being read, or to which the integer result is being written.
- 10⁴⁰. The system of claim ¹31, further comprising a floating point execution unit that operates on floating point data.

45

11
~~41.~~

The system of claim 31 wherein said instruction indicates whether said first register set or said second register set should be used.

~~1242.~~

A method for efficiently utilizing register file resources, comprising the steps of:
(1) executing an instruction having two operands to produce an integer result, said operands are stored in a register file; wherein said register file includes an integer register file and a floating point register file; wherein said instruction indicates the location of said operands;

(2) accessing said integer register file or said floating point register file to retrieve said operands based on said instruction; and

(3) storing said integer result in said integer register file or said floating point register file based on said instruction.

43. A method for efficiently utilizing register file resources, the register set having a first register file configured to store integer data and a second register file configured to store floating point data, comprising the steps of:

1) executing a first instruction to produce a first integer result;

2) storing said integer result in the first register file;

3) executing a second instruction to produce a second integer result; and

4) storing said second integer result in the second register file.--

N
P
B
SUB
DI

46